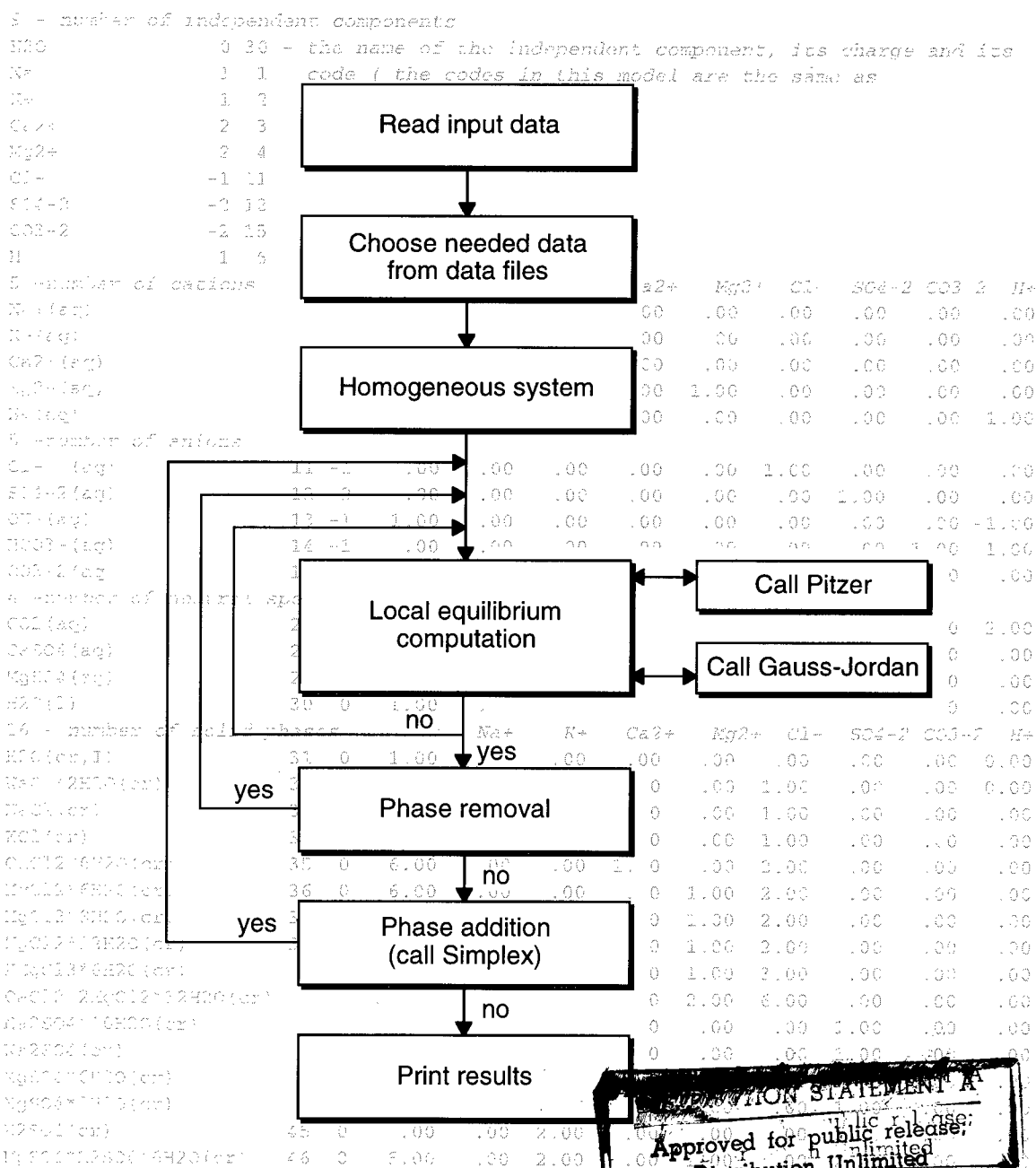


FREZCHEM2

A Chemical Thermodynamic Model for Electrolyte Solutions at Subzero Temperatures

Mikhail V. Mironenko, Steven A. Grant, Giles M. Marion,
and Ronald E. Farren

October 1997



Abstract: This report documents a Fortran version of a chemical thermodynamic model for aqueous electrolyte solutions at subzero temperatures, FREZCHEM2, which is a further development of the FREZCHEM model. The model uses thermodynamic data of Spencer-Møller-Weare that permit the calculation of chemical equilibria in the Na-K-Ca-Mg-Cl-SO₄-H₂O system between -60 and 25°C at atmospheric pressure. It applies the Gibbs energy minimization method for chemical equilibrium computation combined with

Pitzer equations for activity coefficients and water activity calculation. The model includes both the freezing (melting) reaction pathway at fixed water amount and the evaporation (dilution) pathway at fixed temperature. The FREZCHEM2 model can be extended with respect to independent components, electrolyte species, and solids, and if corresponding thermodynamic data are available, the model may be used to compute chemical equilibria in any systems that include aqueous-solution and/or one-component solid phases.

How to get copies of CRREL technical publications:

Department of Defense personnel and contractors may order reports through the Defense Technical Information Center:

DTIC-BR SUITE 0944
8725 JOHN J KINGMAN RD
FT BELVOIR VA 22060-6218
Telephone 1 800 225 3842
E-mail help@dtic.mil
msorders@dtic.mil
WWW http://www.dtic.dla.mil/

All others may order reports through the National Technical Information Service:

NTIS
5285 PORT ROYAL RD
SPRINGFIELD VA 22161
Telephone 1 703 487 4650
1 703 487 4639 (TDD for the hearing-impaired)
E-mail orders@ntis.fedworld.gov
WWW http://www.fedworld.gov/ntis/ntishome.html

A complete list of all CRREL technical publications is available from

USACRREL (CECRL-LP)
72 LYME RD
HANOVER NH 03755-1290
Telephone 1 603 646 4338
E-mail techpubs@crrel.usace.army.mil

For information on all aspects of the Cold Regions Research and Engineering Laboratory, visit our World Wide Web site:
<http://www.crrel.usace.army.mil>

CRREL Report 97-5



**US Army Corps
of Engineers®**

Cold Regions Research &
Engineering Laboratory

FREZCHEM2

A Chemical Thermodynamic Model for Electrolyte Solutions at Subzero Temperatures

Mikhail V. Mironenko, Steven A. Grant, Giles M. Marlon,
and Ronald E. Farren

October 1997

DTIC QUALITY INSPECTED

19971224 025

Prepared for
U.S. ARMY RESEARCH, DEVELOPMENT, AND STANDARDIZATION GROUP (UK)
OFFICE OF THE CHIEF OF ENGINEERS

Approved for public release; distribution is unlimited.

PREFACE

This report was prepared by Dr. Mikhail V. Mironenko, Senior Researcher of the Vernadsky Institute of Geochemistry and Analytical Chemistry of the Russian Academy of Sciences, and Dr. Steven A. Grant, Dr. Giles M. Marion, and Dr. Ronald E. Farren, Research Physical Scientists of the Geochemical Sciences Branch, U.S. Army Cold Regions Research and Engineering Laboratory.

Funding was provided by the European Research Office of the U.S. Army, Project WK2Q6C-7411-EN09. Funding was also provided by U.S. Army Projects AT24-SC-F02, *Chemical Processes in Frozen Soil*, and BT25-EC-B03, *Air-Snow-Ice-Soil Contaminant Interactions in Cold Regions*.

The authors thank Dr. Virgil J. Lunardini of USACRREL and Dr. Jerry P. Greenberg of the University of California at San Diego for reviewing an earlier draft of this manuscript.

The contents of this report are not to be used for advertising or promotional purposes. Citation of brand names does not constitute an official endorsement or approval of the use of such commercial products.

CONTENTS

Preface	ii
Introduction	1
Mathematical algorithm	1
Local minimum computation	1
Search for the equilibrium phase assemblage	4
Special steps	4
FREZCHEM2 program	4
Data files	5
Program input and output	5
Literature cited	6
Appendix A: Fortran listing of the FREZCHEM2 program	11
Appendix B: Data files for program FREZCHEM2	39
Abstract	41

ILLUSTRATION

Figure	
1. Flowchart of the FREZCHEM2 model	5

TABLES

Table	
1. FREZCHEM2 model input for freezing seawater from 0° down to -40°C with a 2°C decrement	6
2. FREZCHEM2 model output for freezing seawater at -45°C and at -55°C	7
3. FREZCHEM2 model output for evaporation of seawater at 0°C	8
4. Temperatures of first appearance of solid phases on chilling seawater	8

FREZCHEM2

A Chemical Thermodynamic Model for Electrolyte Solutions at Subzero Temperatures

MIKHAIL V. MIRONENKO, STEVEN A. GRANT, GILES M. MARION, AND RONALD E. FARREN

INTRODUCTION

The FREZCHEM model was developed by Marion and Grant (1994) to calculate chemical equilibria among aqueous electrolyte solutions, ice, and salts. The model applies the Pitzer equations for calculation of aqueous species and water activities. To find chemical equilibrium, this program solves sequentially a set of nonlinear equations that includes both solid-phase deposition and ion-pair formation using an individual subroutine for every reaction. FREZCHEM uses data on constants of chemical reactions and Pitzer equation parameters published by Spencer et al. (1990). The results of modeling show good agreement both with experimental data and with the results of the Spencer-Møller-Weare model. However, the FREZCHEM model has some limitations. One is convergence problems at high ionic strengths (>15 molal) and at junctions, where new phases begin to precipitate. Another is that addition of any new substance into this model requires changes not only in data but also in the program code.

The objective of this report is the further development of the chemical thermodynamic model FREZCHEM to make it more reliable, universal, and flexible. The point calculation reliability was improved by applying the Gibbs energy minimization approach to computing equilibrium. The thermodynamic information needed for computations is separated from the calculating routines. That allows components to be added to the system without code changes and the program code to be applied for other chemical systems. It should be noted that the Pitzer approach describes most interactions in aqueous solution as electro-

static and only explicitly recognizes a few chemical interactions, such as ion-pair formation. This is why the system under consideration is very simple from the viewpoint of chemical interactions, but it is very complex from the viewpoint of the influence of activity coefficients on the behavior of the Gibbs energy function.

MATHEMATICAL ALGORITHM

The system under investigation consists of the following components: 1) solid salts of fixed chemical composition and pure ice (so-called one-component phases), and 2) aqueous solutions consisting of water and dissolved electrolytes. The applied algorithms of chemical equilibria computation will be described in terms of these components.

The equilibrium composition of the system at constant T , P , and specified bulk composition may be found by minimizing the Gibbs energy function of the system under balance restrictions.

Local minimum computation

Local minimum is considered as an equilibrium composition of the system, in which all existing phases are specified before computation. The Gibbs energy function of the system that contains M solids and aqueous solution (water and J species) is as follows:

$$g = \frac{G}{RT} = \sum_{k=1}^M \mu_k^0 n_k + \mu_w n_w + \sum_{j=1}^J \mu_j n_j$$

where G = free energy of the system,
 n = the molal quantity of components,

μ_j = the chemical potential of species j ,
 μ_w = the chemical potential of water,
 μ_k^0 = the standard chemical potential of a
 one-component solid-phase k , and
 R = the universal gas constant.

The chemical potential of aqueous solution species j in terms of molality is defined by

$$\mu_j = \mu_j^0 + \ln a_j = \mu_j^0 + \ln(m_j \gamma_j)$$

where μ^0 is the standard chemical potential,

$$m_j = \frac{n_j}{n_w} \cdot 55.51$$

is the moles of the j^{th} species per 1 kg of water (molality), and γ is an activity coefficient.

The chemical potential of water may be written as

$$\mu_w = \mu_w^0 + \ln a_w$$

where water activity a_w , according to Pitzer (1987), is defined through the osmotic coefficient of the solution ϕ and molalities of species by

$$\ln a_w = -\phi \frac{W}{1000} \left(\sum_j m_j \right)$$

where W is the molecular weight of water (18.0153).

Accordingly, the free energy function of the system is as follows:

$$\begin{aligned}
 g(\vec{n}) = & \sum_{k=1}^M \mu_k^0 n_k + n_w (\mu_w^0 - \phi \frac{\sum n_j}{n_w}) \\
 & + \sum_{j=1}^J n_j \left[\mu_j^0 + \ln \left(\frac{n_j 55.51}{n_w} \gamma_j \right) \right]. \quad (1)
 \end{aligned}$$

Mass balance constraints, including the electroneutrality equation if necessary, may be written as a system of linear equations:

$$\sum_{j=1}^{M+1+J} v_{ij} n_j = b_i, \quad i = 1, P \quad (2)$$

where P is the number of independent chemical components in the system, and v_{ij} is the number of moles (stoichiometric units) of independent component i in one mole of component j . b_i represents the number of moles of independent com-

ponent i in the system. For the electroneutrality equation $b_i = 0$ and $v_{ij} = z_j$, where z_j is the charge of the j^{th} component. In matrix notation, eq 2 may be written as

$$N \vec{n}^T = \vec{b}$$

where N is the stoichiometric matrix, \vec{n} is the vector of numbers of moles of species, and \vec{b} is the vector of bulk chemical composition of the system.

It is convenient to solve the system of linear equations (eq 2) with respect to P components, including M ($M \leq P$) solids,

$$n_k = B_k - \sum_{j=1}^{J+1} a_{kj} n_j \quad k = 1, M \quad (2a)$$

$$f_j = B_j - \sum_{j=1}^{J+1} a_{ij} n_j = 0 \quad i = M+1, P \quad (2b)$$

and in this way to switch to new independent components. In such a manner the stoichiometry of other $J+1-P$ components (vectors \vec{a}_j) and the matter balance \vec{B} are now defined through these independent components. This operation allows the number of active constraints to be reduced up to $P-M$. The thermodynamic meaning of this lies in the fact that the chemical potential of a one-component solid phase is equal to the standard Gibbs energy of formation and does not depend on its amount, until this phase is present. This is why the system can be considered to be open with respect to this component.

It is obvious that

$$n_w > 0 \text{ and } n_j > 0. \quad (3)$$

Minimization of the function in eq 1 under the constraints of eq 2 and eq 3 can be replaced by a search of the extremum of the Lagrangian function, which may be written as

$$\begin{aligned}
 \Phi(\vec{n}, \vec{\lambda}) = & g(\vec{n}) + \sum_{k=1}^M \mu_k^0 (B_k - \sum_{j=1}^{J+1} a_{kj} n_j) \\
 & + \sum_{i=M+1}^P \lambda_i (B_i - \sum_{j=1}^{J+1} a_{ij} n_j)
 \end{aligned}$$

where λ is a Lagrangian multiplier. It can be shown (Karpov et al. 1976) that λ is the chemical potential of the corresponding independent component of the system. In particular for solids, $\lambda_k = \mu_k^0$.

The conditions of extremum of the Lagrangian function are found where all first partial derivatives with respect to components and to Lagrangian multipliers are equal to zero. This gives

$$\frac{\partial \Phi}{\partial n_j} = (\mu_j^0 + \ln 55.51) + \ln\left(\frac{n_j}{n_w} \gamma_j\right) - \sum_{k=1}^M \mu_k^0 a_{kj} - \sum_{i=M+1}^P \lambda_i a_{ij} = 0 \quad (4a)$$

$$\frac{\partial \Phi}{\partial n_w} = -\phi \frac{n_j}{n_w} - \sum_{k=1}^M \mu_k^0 a_{kw} - \sum_{i=M+1}^P \lambda_i a_{iw} = 0 \quad (4b)$$

$$\frac{\partial \Phi}{\partial \lambda_i} = B_i - \sum_{j=1}^J a_{ij} n_j - a_{iw} n_w = 0. \quad (4c)$$

There are different approaches to searching the Lagrangian function extremum. One of them is the algorithm developed by White (1958, 1967) for homogeneous gas systems, which has been further developed for heterogeneous multiphase systems by Karpov (1976). This algorithm was applied by the senior author of this report for computation of a wide range of chemical equilibria (Mironenko 1991, 1992) and is build into the DiaNIK system (Khodakovskiy 1992). The idea of the method, as applied to the system under consideration, is as follows. Equation 4a may be solved with respect to n_j :

$$n_j = \frac{n_w}{\gamma_j} \exp\left(\sum_{k=1}^M \mu_k^0 a_{kj} + \sum_{i=M+1}^P \lambda_i a_{ij} - \mu_j^0\right). \quad (5)$$

Substitution of these terms into eq 4b and c gives the system of $P - M + 1$ equations, which may be solved by Newton's method for λ and n_w . The advantages of this approach are a fast rate of computation, due to a small amount of variables (their amount does not depend on the number of species), and the lack of necessity to undertake special steps to calculate species at very low concentrations or to correct negative values of mass for species during iterations. Unfortunately, attempts to apply this approach to brine systems in combination with Pitzer's routine have demonstrated that the algorithm is not tolerant of oscillations of activity coefficient values, provided

by Pitzer's routine at every iteration. Because of this it was very difficult to reach the required precision of solution (0.1%), even when special steps were undertaken.

Another approach is to solve the whole system of $P - M + J + 1$ equations (eq 4 a,b,c) iteratively by Newton's method for n_j , n_w , and λ_k . This algorithm has been described in detail by Harvie et al. (1987). It has been successfully applied by Spencer et al. (1990) for strong electrolyte solution modeling, but a working version of the program has not been published. This algorithm also was applied by Mironenko (1983) for modeling fluid-rock interactions during hydrothermal uranium ore formation.

The second partial derivatives of the Lagrangian function are equal to

$$\frac{\partial^2 \Phi}{\partial n_j \partial n_{j1}} = \begin{cases} \frac{1}{n_j}, & \text{if } j = j1 \\ 0, & \text{if } j \neq j1 \end{cases}$$

$$\frac{\partial^2 \Phi}{\partial n_j \partial n_w} = -\frac{1}{n_w}$$

$$\frac{\partial^2 \Phi}{\partial n_j \partial \lambda_i} = \frac{\partial^2 \Phi}{\partial \lambda_i \partial n_j} = -a_{ij}$$

$$\frac{\partial^2 \Phi}{\partial n_w \partial n_j} = -\phi \frac{1}{n_w}$$

$$\frac{\partial^2 \Phi}{\partial n_w^2} = \frac{\sum n_j}{n_w^2} \phi$$

$$\frac{\partial^2 \Phi}{\partial n_w \partial \lambda_i} = \frac{\partial^2 \Phi}{\partial \lambda_i \partial n_w} = -a_{iw}$$

$$\frac{\partial^2 \Phi}{\partial \lambda_i \partial \lambda_{i1}} = 0.$$

This matrix of second partial derivatives is known as the Hessian matrix.

Activity coefficients of species and the osmotic coefficient are calculated at every iteration using Pitzer's model (Pitzer 1987). In FREZCHEM2, the Pitzer routines published by Marion and Grant (1994) were used, with insignificant changes dealing mainly with the interface with data files.

The molal amounts of solids are calculated after a local equilibrium has been achieved using eq 2a.

Search for the equilibrium phase assemblage

Solids

If a molal quantity of solid calculated using eq 2a was negative, this solid phase was considered as completely dissolved, and a new local equilibrium without this solid was computed. Then a search of new solid phases from the list of possible solids to be included into the system is undertaken. The criterion for the inclusion of phase K is as follows:

$$\sum_{k=1}^M \mu_k^0 a_{kK} + \sum_{i=M+1}^P \lambda_i a_{iK} - \mu_K^0 < 0. \quad (6)$$

The thermodynamic meaning of this expression is that the free energy of chemical reaction of a given solid substance formed from independent components of the system is negative, and therefore this solid is thermodynamically stable. If the condition (eq 6) asserts, this solid replaces one of the independent components of the system with which it is linearly dependent. Then the system of linear equations (eq 2a,b) is solved with respect to this new independent component. By this means the chemical composition of all components of the system will be expressed in terms of this and other independent components. This procedure is largely achieved by applying the Simplex routine. (Simplex is a classic finite iteration method of linear programming [Korn and Korn 1963].) Addition of each solid phase reduces the number of active linear restrictions by one. Calculations are continued until, in the list of possible phases, there is no phase that meets the condition in eq 6.

Aqueous solution

Aqueous solution is considered absent in the system when the number of active balance restrictions ($P - M$) is less than or equal to one and the amount of water is less than 0.001 moles.

Special steps

Usually, the approximate phase composition of a system may be determined at the first steps of calculation using the Simplex routine. Then the exact equilibrium composition may be computed using Newton's method for local equilibria determination and Simplex methods for addition or substitution of solids. Due to the very high non-ideality of brines, this technique collapsed, and some changes in the logical pattern of calculation were made:

1. At first, the system is considered homogeneous (no solids), then the solid-phase assem-

blage is calculated, not simultaneously during one application of the Simplex routine, but sequentially. Another phase is added after local equilibrium with previously added phases is achieved.

2. After the appearance of a new solid phase and before applying Newton's method, the current species concentrations have to be recalculated to be in better agreement with values of independent component chemical potentials. The relation between concentration of species and values of chemical potentials of independent components is expressed by eq 5 and can be also treated in terms of the free energy of the chemical reaction of species formation from independent components of the system.

3. Because of particularities of Pitzer's model, to prevent wide fluctuations during solution of the system of equations 4a,b,c by Newton's method, we have to smooth changes of activity coefficient and osmotic coefficient values, which are calculated at each iteration by Pitzer's routine. We use average values obtained at the current and previous iterations.

4. At every iteration a new approximation to the solution is provided by inversion of the Hessian matrix: $y_i^{(k+1)} = y_i^{(k)} + \Delta_i / \xi$. For a homogeneous system the value of ξ is equal to 1 and it increases by 0.5 with every new solid that precipitates.

FREZCHEM2 PROGRAM

A listing of the FREZCHEM2 Fortran program is in Appendix A. FREZCHEM2 consists of a main program called READWRITE and seven subroutines.

The READWRITE program reads input data from the file INPUT, according to these data forms independent components of the system, and reads the temperature interval and temperature step for freezing, or the water content interval and water decrement at a given temperature for the evaporation scenario. It calculates chemical potentials of the components as functions of temperature, calls various subroutines, and writes results of the chemical equilibria computation into the file RESULT.

Subroutine CHOICE is called from the main program and chooses components that may be formed in the system of given chemical composition as well as their stoichiometry. A data file for this routine is the DATABASE file.

Subroutine SOL is called from READWRITE

and is the main calculating routine. It computes the equilibrium composition of the system at given T and specified mass balance by searching for the global extremum of the Lagrangian function. It forms the Hessian matrix and calls various subroutines. Results of computations return to the READWRITE program to be written.

Subroutine SIMPL is called from the SOL subroutine to enter new solid phases into the system and to invert the stoichiometric matrix. This calls the GG1 routine, which is the short version of the Gauss-Jordan matrix method.

Subroutine PITZER is called from the main program to choose data for Pitzer parameters and to calculate them at various temperatures if the freezing scenario has been chosen. It is also called from the SOL routine to calculate activity coefficients of species and water activity, using the Pitzer model at every iteration while minimizing the free energy function by Newton's method. In FREZCHEM2, the PITZER subroutine, as well as the INTERACT subroutine published by Marion and Grant (1994), were used with only insignificant changes dealing mainly with the interface with data files.

Subroutine INTERACT calculates the higher-order electrostatic interactions for the Pitzer equations.

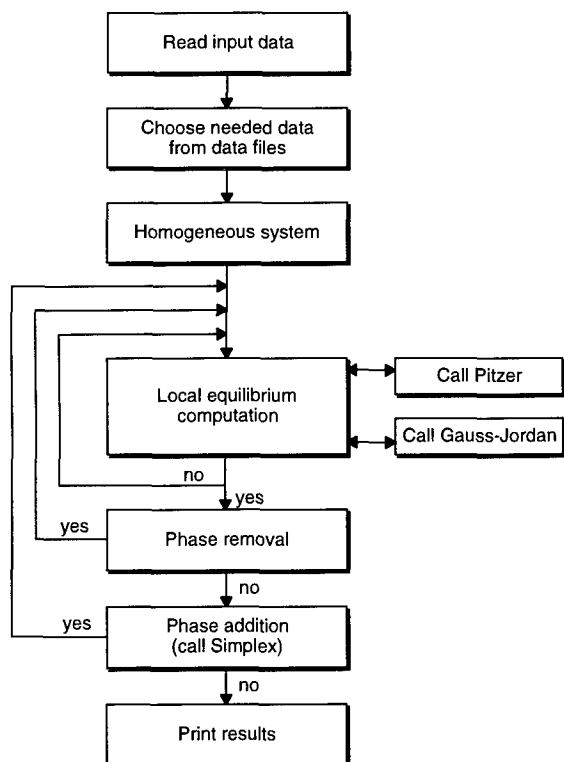


Figure 1. Flowchart of the FREZCHEM2 model.

Subroutine GG is called from the SOL subroutine and solves the system of linear equations by the Gauss-Jordan method.

The principal flowchart of FREZCHEM2 is shown in Figure 1. The FREZCHEM2 model is a universal model that may be used to compute chemical equilibrium in any system consisting of one-component solids and/or aqueous solution. For this goal, only additions in the data files are needed.

Data files

Files that contain the information needed for calculations and some remarks are listed in Appendix B.

File DATABASE contains a list of independent components, which can be taken into account, and lists of aqueous solution species (cations, anions, neutral species including water) and solid phases, consisting of given independent components as well as their stoichiometry. For the convenience of users, the same species numbers (coding) were used as in the FREZCHEM model. If necessary, additional information may be added for independent components as well as for species and solids. Requested formats for entering new data could be taken from the listing of the READWRITE program (Appendix A).

File TABLE1 represents Table 1 of Spencer et al. (1990), which includes constants for the Debye-Hückel model parameter A^ϕ and for the binary interaction parameters as a function of temperature (K). File TABLE2 represents Table 2 of that paper for mixed-salt parameters. File TABLE3 contains coefficients for calculation of free energies of chemical reactions of formation for solids and ion pairs from aqueous solution species and liquid water as a function of temperature, using equations of the form published by Spencer et al. (1990):

$$\begin{aligned} \frac{-\Delta G}{RT} = & a_1 + a_2T + a_6T^2 + a_9T^3 \\ & + a_3/T + a_4 \ln(T) \end{aligned} \quad (7)$$

and represents a copy of Table 3 from their paper. In this convention, free energies of cations, anions, and liquid water are taken to be equal to zero at any temperature.

Program input and output

Input to FREZCHEM2 is through the file INPUT, which contains the molal amounts of inde-

Table 1. FREZCHEM2 model input for freezing seawater from 0°C down to -40°C with a 2°C decrement.

<i>SMW seawater</i>	<i>Title of the task</i>
0.48695	Sodium (mol/kg)
0.01063	Potassium (mol/kg)
0.00953	Calcium (mol/kg)
0.05516	Magnesium (mol/kg)
0.56818	Chloride (mol/kg)
0.02939	Sulfate (mol/kg)
0.0	Carbonate (mol/kg)
0.0	Hydrogen (mol/kg)
273.15	<i>initial temperature</i>
1	<i>freezing (2 for evaporation)</i>
233.15	<i>final temperature (final amount of water for evaporation)</i>
2.0	<i>temperature decrement (water decrement for evaporation)</i>

pendent components (presently Na⁺, K⁺, Ca⁺², Mg⁺², Cl⁻, SO₄⁻², and CO₃⁻² are included) per 1 kg of water. Thus, an initial amount of water in the system is equal to 1 kg or 55.51 moles. If the molal quantity of an independent component is equal to zero, all substances in the file DATABASE that contain it will be ignored. The program calculates a charge balance and, if it is different from zero, proposes to add some amount of any cation or anion depending on the value and the sign of the charge balance, after which the program starts its work. FREZCHEM2 is able to calculate both 1) a cooling/heating scenario (in this case one enters initial and final temperatures and temperature decrement/increment) and 2) an evaporation/dilution scenario at constant temperature (in this case it needs a temperature, a water amount decrement/increment, and a final mass of water). The model is also able to calculate the process of ice evaporation. An example of input (file INPUT) is present in Table 1 for seawater freezing from 0 to -40°C for a 2°C temperature decrement. The explanations are written in italics.

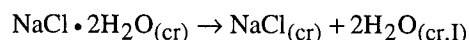
Output from the program is to the RESULT file. As examples of output, results of computation of seawater freezing at 228.15 K (-45°C) and at 218.15 K (-55°C) are given in Table 2. Output of evaporation of seawater at 273.15 K (0°C) down to 50 g of water is given in Table 3.

Distribution of the independent components among solids and solution phases is given in the BALANCE table at the bottom of the output table. The last column of this table has been printed to show the equivalence of input and computed balances.

Verification of the model

To verify the program, phase diagrams from Spencer et al. (1990) and point computations from Marion and Grant (1994) were recalculated. The model reproduces these computations with good accuracy. Table 4 shows the temperatures at the appearance of solids during seawater freezing, taken from Spencer et al. (1990) with an added column obtained by the FREZCHEM2 model using their thermodynamic data.

It is interesting to note that, according to the free energies of chemical reactions in the model, a solid reaction



takes place at temperatures lower than -57.15°C. To verify this independently, the heat capacity equations for these phases at low temperatures are needed.

LITERATURE CITED

- Fujino, K., E.L. Lewis, and R.G. Perkin (1974) The freezing points of seawater at pressures up to 100 bars. *Journal of Geophysical Research*, **79**: 1792-1797.
- Harvie, C.E., J.P. Greenberg, and J.H. Weare (1987) A chemical equilibrium algorithm for highly non-ideal multiphase system: Free energy minimization. *Geochimica et Cosmochimica Acta*, **51**: 13-21.
- Karpov, I.K., A.V. Kiselev, and F.A. Letnikov (1976) *Computer modeling of natural mineral formation*. (In Russian.) Moscow: Nedra.
- Khodakovsky, I.L. (1992) DiaNIK—Thermodynamic databases of minerals and calculations of equilibrium compositions of natural systems. In *Abstracts of the 12th IUPAC Conference on Chemical Thermodynamics, Snowbird, Utah, U.S.A.*, p. 157-158.
- Korn, G.A., and T.M. Korn (1963) *Mathematical Handbook for Scientists and Engineers*. New York: McGraw-Hill Book Company.
- Marion, G.M., and S.A. Grant (1994) FREZCHEM: A chemical thermodynamic model for aqueous solutions at subzero temperatures. USA Cold Regions Research and Engineering Laboratory, Special Report 94-18.
- Mironenko, M.V. (1983) Physical-chemical modeling of uranium ore formation in the uranium deposit [in Russian]. Ph.D. dissertation (unpublished), Vernadsky Institute of Geochemistry and Analytical Chemistry, Moscow.
- Mironenko, M.V., and A.N. Salaskin (1991) PTX-

Table 2. FREZCHEM2 model output for freezing seawater at -45°C and at -55°C.

Seawater freezing
temperature -45.00°C (228.15 K)

SOLID PHASES

<i>N</i>	<i>Phase</i>	<i>Moles</i>	<i>-G/RT</i>
1	H ₂ O _(cr,I)	53.52785	-0.4278
2	NaCl*2H ₂ O _(cr)	0.42624	1.0995
3	KCl _(cr)	0.00948	-0.8002
4	MgCl ₂ *12H ₂ O _(cr)	0.05052	1.2364
5	Na ₂ SO ₄ *10H ₂ O _(cr)	0.02925	-12.2171

AQUEOUS SOLUTION

Ionic strength 11.0759
Osmotic coefficient 2.0008

<i>N</i>	<i>Species</i>	<i>Moles</i>	<i>Molality</i>	<i>Activity</i>	<i>Act. coef.</i>	<i>-G/RT</i>
1	Na ⁺ _(aq)	0.22059D-02	0.53403D+00	0.23599D+00	0.4419	0.0000
2	K ⁺ _(aq)	0.11533D-02	0.27920D+00	0.15016D-01	0.0538	0.0000
3	Ca ⁺²	0.94653D-02	0.22914D+01	0.15724D+01	0.6862	0.0000
4	Mg ⁺²	0.46407D-02	0.11235D+01	0.65163D+00	0.5800	0.0000
5	Cl ⁻ _(aq)	0.31425D-01	0.76077D+01	0.29938D+02	3.9352	0.0000
6	SO ₄ ⁻² _(aq)	0.72974D-04	0.17666D-01	0.64149D-02	0.3631	0.0000
7	CaSO _{4(aq)}	0.64693D-04	0.15662D-01	0.15662D-01	1.0000	-0.4410
8	MgSO _{4(aq)}	0.87333D-07	0.21142D-04	0.21142D-04	1.0000	5.2851
9	H ₂ O(l)	0.22929D+00		0.65191D+00		0.0000

BALANCE

	<i>Total</i>	<i>Solids</i>	<i>Solution</i>	<i>Total computed</i>
H ₂ O	0.555084E+02	0.552791E+02	0.229289E+00	0.555084E+02
Na ⁺	0.486950E+00	0.484744E+00	0.220592E-02	0.486950E+00
K ⁺	0.106300E-01	0.947671E-02	0.115329E-02	0.106300E-01
Ca ⁺²	0.953000E-02	0.000000E+00	0.953000E-02	0.953000E-02
Mg ⁺²	0.551600E-01	0.505192E-01	0.464076E-02	0.551600E-01
Cl ⁻	0.568180E+00	0.536755E+00	0.314254E-01	0.568180E+00
SO ₄ ⁻²	0.293900E-01	0.292522E-01	0.137754E-03	0.293900E-01

Number of iterations 93

Temperature -55.00°C (218.15 K)

SOLID PHASES

<i>N</i>	<i>Phase</i>	<i>Moles</i>	<i>-G/RT</i>
1	H ₂ O _(cr,I)	53.63903	-0.5004
2	NaCl*2H ₂ O _(cr)	0.42817	0.8123
3	KCl _(cr)	0.01063	-1.4388
4	CaCl ₂ *6H ₂ O _(cr)	0.00953	5.6614
5	MgCl ₂ *12H ₂ O _(cr)	0.05516	-1.5678
6	Na ₂ SO ₄ *10H ₂ O _(cr)	0.02939	-15.0450

BALANCE

	<i>Total</i>	<i>Solids</i>	<i>Solution</i>	<i>Total computed</i>
H ₂ O	0.555084E+02	0.555084E+02	0.000000E+00	0.555084E+02
Na ⁺	0.486950E+00	0.486950E+00	0.000000E+00	0.486950E+00
K ⁺	0.106300E-01	0.106300E-01	0.000000E+00	0.106300E-01
Ca ⁺²	0.953000E-02	0.953000E-02	0.000000E+00	0.953000E-02
Mg ⁺²	0.551600E-01	0.551600E-01	0.000000E+00	0.551600E-01
Cl ⁻	0.568180E+00	0.568180E+00	0.000000E+00	0.568180E+00
SO ₄ ⁻²	0.293900E-01	0.293900E-01	0.000000E+00	0.293900E-01

Number of iterations 88

Table 3. FREZCHEM2 model output for evaporation of seawater at 0°C.

Temperature	0.00°C (273.15 K)					
Water amount	50.00 g					
SOLID PHASES						
<i>N</i>	<i>Phase</i>	<i>Moles</i>	<i>-G/RT</i>			
1	NaCl _(cr)	0.28103	3.4448			
2	Na ₂ SO ₄ *10H ₂ O _(cr)	0.02021	-5.7260			
AQUEOUS SOLUTION						
Ionic strength		8.0709				
Osmotic coefficient		1.5573				
<i>N</i>	<i>Species</i>	<i>Moles</i>	<i>Molality</i>	<i>Activity</i>	<i>Act. coef.</i>	<i>-G/RT</i>
1	Na ⁺ _(aq)	0.16551D+00	0.35701D+01	0.28612D+01	0.8014	0.0000
2	K ⁺ _(aq)	0.10630D-01	0.22929D+00	0.74428D-01	0.3246	0.0000
3	Ca ⁺²	0.82305D-02	0.17754D+00	0.16740D+00	0.9429	0.0000
4	Mg ⁺²	0.55155D-01	0.11897D+01	0.27250D+01	2.2905	0.0000
5	Cl ⁻ _(aq)	0.28715D+00	0.61941D+01	0.10952D+02	1.7681	0.0000
6	SO ₄ ⁻² _(aq)	0.78786D-02	0.16995D+00	0.10190D-01	0.0600	0.0000
7	CaSO _{4(aq)}	0.12993D-02	0.28027D-01	0.28027D-01	1.0000	-2.7979
8	MgSO _{4(aq)}	0.53787D-05	0.11602D-03	0.11602D-03	1.0000	5.4788
9	H ₂ O(l)	0.25733D+01		0.72306D+00	0.0000	
BALANCE						
	<i>Total</i>	<i>Solids</i>	<i>Solution</i>	<i>Total computed</i>		
H ₂ O	0.277542E+01	0.202067E+00	0.257334E+01	0.277541E+01		
Na ⁺	0.486950E+00	0.321441E+00	0.165509E+00	0.486950E+00		
K ⁺	0.106300E-01	0.000000E+00	0.106300E-01	0.106300E-01		
Ca ⁺²	0.953000E-02	0.000000E+00	0.952987E-02	0.952987E-02		
Mg ⁺²	0.551600E-01	0.000000E+00	0.551600E-01	0.551600E-01		
Cl ⁻	0.568180E+00	0.281028E+00	0.287154E+00	0.568182E+00		
SO ₄ ⁻²	0.293900E-01	0.202067E-01	0.918331E-02	0.293900E-01		
Number of iterations		45				

Table 4. Temperatures (°C) of first appearance of solid phases on chilling of seawater.

<i>Solid</i>	<i>Experiment</i>	<i>FREZCHEM2</i>	
		<i>Model^a</i>	<i>model</i>
Ice	-1.921 ^b	-1.924	-1.921
Mirabilite	-8.2 ^c	-5.90	-5.87
Hydrohalite	-22.9 ^c	-22.84	-22.87
Sylvite	-36 ^c	-34.25	-34.30
MgCl ₂ · 12H ₂ O	-36 ^c	-36.82	-36.82
Antarcticite	-54 ^c	-53.64 ^d	-53.73

a. Spencer et al. (1990).

b. Fujino et al. (1974).

c. Nelson and Thompson (1954).

d. Calculated in sulfate-free system.

determination, computer thermodynamic model of fluid-rock interaction and uranium deposition. In *Source, Transport and Deposition of Metals, Proceedings of the 25-Year SGA Anniversary Meeting, Nancy, France*. Balkema, p. 79–81.

Mironenko, M.V., M.Yu. Zolotov, and M.Yu. Frenkel (1992) Algorithm, computer code and data base for computation equilibria in systems involving solid, aqueous and gas nonideal solutions. In *Abstracts of the 2nd International Symposium "Thermodynamics of Nature Processes," 14–20 September, Novosibirsk*. United Institute of Geology, Geophysics, and Mineralogy, Siberian Branch of the Russian Academy of Sciences, p. 117.

Nelson, K.H., and T.G. Thompson (1954) Deposition of salts from sea water by frigid concentration. *Journal of Marine Research*, **13**(2): 166–182.

Pitzer, K.S. (1987) A thermodynamic model for aqueous solutions of liquid-like density. *Reviews in Mineralogy*, **17**: 97–142.

Spencer, R.J., N. Møller, and J.H. Weare (1990) The prediction of mineral solubility in natural waters: A chemical equilibrium model for the Na-K-Ca-Mg-Cl-SO₄-H₂O system at temperatures below 25°C. *Geochimica et Cosmochimica Acta*, **54**: 575–590.

White, W.B. (1967) Numerical determination of chemical equilibrium and partitioning of free energy. *Journal of Chemical Physics*, **46**: 4173–4175.

White, W.B., S.M. Johnson, and G.B. Datzig (1958) Chemical equilibrium in complex mixtures. *Journal of Chemical Physics*, **28**: 751–755.

APPENDIX A: FORTRAN LISTING OF THE FREZCHEM2 PROGRAM

```

PROGRAM READWRITE
IMPLICIT DOUBLE PRECISION (A-H,O-Z)

CHARACTER *40 NMC,NEL,NM,NMCOMP,TITLE

DIMENSION BAL(10),NEL(10),G0(40,6),G(40),GIN(40),BALS(10)
DIMENSION BALL(10),ACT(20)
DIMENSION BC(10,20),A(10,40),EX(40),NMC(20),NM(40),AIN(10,40)
DIMENSION IZC(20),IZ(20),NUM(40),NUMC(20),NUMIN(40),NMM(40)

COMMON /COMPNT/ NUR(10),IP,NUL(10),IPNUL,KCOMP
COMMON /NUMBER/ NCAT,NANI,NNEI,NMM
COMMON /MASS1/ A,G,NUM,X(40)
COMMON /MASS2/ NM
COMMON /INTGER/ NL,N,N1
COMMON /MATRIX/ ACT,UM,PHI,AH2O,IZ,IPP,NMV
COMMON /AIN/ AIN,GIN,T,NUMIN,IV
COMMON /OUTPUT/ JOPEN(10),IOPEN

10  FORMAT(A40)
11  FORMAT(A40/)
12  FORMAT(I3/)
13  FORMAT(A15,I3,1X,I2,A40,F12.6)
14  FORMAT(' THE CHARGE OF 1 KG OF THE SOLUTION IS EQUAL TO
$,F8.5)
15  FORMAT(/13X,'Temperature ',F7.2,' C (' ,F7.2,' K)')
16  FORMAT (' WATER AMOUNT ',F7.2,' GRAM' /
$ ' _____')
17  FORMAT(25X,'SOLID PHASES'/1X,'N Phase ',
$ 16X,' Moles -G/RT')
18  FORMAT(I2,1X,A23,F9.5,2X,F12.4)
19  FORMAT(21X,'Ionic strength ',F8.4/
$ 21X,'Osmotic coefficient ',F7.4)
20  FORMAT(1X,'N',2X,'Species ',4X,'Moles',7X,
$ 'Molality ',3X,'Activity ', 'Act.coef. -G/RT')
21  FORMAT(I2,1X,A16,3D12.5,2X,F8.4,F10.4)
22  FORMAT(I2,1X,A16,D12.5,12X,D12.5,8X,F10.4,)
23  FORMAT (26X,'BALANCE'/
$ 13X,'Total',9X,'Solids',8X,'Solution',5X,'Total computed')
24  FORMAT(1X,A8,4(2X,E12.6))
25  FORMAT(' Number of iterations ',I3)
26  FORMAT(' THE PROGRAM CANNOT CALCULATE THIS CHEMICAL
$EQUILIBRIUM')

OPEN(1,FILE='DBASE')
OPEN(9,FILE='INPUT')
OPEN(10,FILE='RESULT')

XW=55.50837

READ(9,10) TITLE
WRITE(10,11) TITLE

```

```

C..  READING  INDEPENDENT COMPONENTS AND GIVING THE BALANCE

      READ(1,12) KCOMP

C..  THE WATER BALANCE (1 KG H2O = 55.50837 MOLES )

      BAL(1)=XW
      NEL(1)='H2O'
      NUMIN(1)=1
      NMM(1)=30
      NM(1)=NEL(1)
      NUR(1)=1
      J=1
      J1=0
      CHARGE=0.

      DO 2000 I=2,KCOMP
        READ(1,13) NMCOMP,IIZ,NU
        READ(9,*) AA
        IF (AA.GT.1.D-20) THEN
          J=J+1
          BAL(J)=AA
          CHARGE=CHARGE+IIZ*AA
          NEL(J)=NMCOMP
          NM(J)=NEL(J)
          NMM(J)=NU
          NUMIN(J)=J
          NUR(J)=I
        ELSE
          J1=J1+1
          NUL(J1)=I
        END IF
      2000  CONTINUE

      IF(DABS(CHARGE).GT.0.00001) THEN
        WRITE(*,*) 'THE SALT BALANCE IS NOT CORRECT'
        WRITE(*,14) CHARGE
        IF(CHARGE.GT.0.)WRITE(*,*)'ADD ANIONS OR SUBT. CATIONS IN
$INPUT'
        IF(CHARGE.LT.0.)WRITE(*,*)'ADD CATIONS OR SUBT. ANIONS IN
$INPUT'
        PAUSE
        GO TO 815
      END IF

      IP=J
      IPP=IP
      IPNUL=J1

      DO 2100 I=1,IP
        DO 2101 J=1,IP
          AIN(I,J)=0.
        2101  CONTINUE

          AIN(I,I)=1.
          GIN(I)=0.
        2100  CONTINUE

```



```

      CALL CHOICE (NCAT,BC,NMC,NUMC,IZC)
      NC=IP+NCAT

      DO 2200 I=1, NCAT
        I1=I+IP

        DO 2201 J=1, IP
          AIN(J,I1)=BC(J,I)
2201      CONTINUE

        NM(I1)=NMC(I)
        NMM(I1)=NUMC(I)
        NUMIN(I1)=I1
        IZ(I1)=IZC(I)
2200      CONTINUE

      CALL CHOICE (NANI,BC,NMC,NUMC,IZC)
      NA=NC+NANI

      DO 2300 I=1, NANI
        I1=I+NC

        DO 2301 J=1, IP
          AIN(J,I1)=BC(J,I)
2301      CONTINUE

        NM(I1)=NMC(I)
        NMM(I1)=NUMC(I)
        NUMIN(I1)=I1
        IZ(I1)=IZC(I)
2300      CONTINUE

      CALL CHOICE (NNEI,BC,NMC,NUMC,IZC)
      NL=NA+NNEI

      DO 2400 I=1, NNEI
        I1=I+NA

        DO 2401 J=1, IP
          AIN(J,I1)=BC(J,I)
2401      CONTINUE

        NM(I1)=NMC(I)
        NMM(I1)=NUMC(I)
        IF(NMM(I1).EQ.30) IV=I1
        NUMIN(I1)=I1
        IZ(I1)=IZC(I)
2400      CONTINUE

      CALL CHOICE (NSOL,BC,NMC,NUMC,IZC)

      DO 2500 I=1, NSOL
        I1=I+NL

        DO 2501 J=1, IP
          AIN(J,I1)=BC(J,I)
2501      CONTINUE

```

```

        NM(I1)=NMC(I)
        NMM(I1)=NUMC(I)
        NUMIN(I1)=I1
2500  CONTINUE

        IP1=IP+1
        L=NCAT+NANI+NNEI
        N=NL+NSOL
        N1=N+1

        DO 2600 J=1,IP
            AIN(J,N1)=BAL(J)
2600  CONTINUE

        IJJ=0

        DO 2700 I=1, N1
            NUM(I)=NUMIN(I)
2700  CONTINUE

C..... READ TABLE 3 .....

        OPEN(5, FILE='TABLE3')

        DO 2800 I=IP+1,N
2810      READ(5,*, END=2820) NT1

            IF(NMM(I).EQ.NT1) THEN
                BACKSPACE(5)
                READ(5,*) NT1, (G0(I,K),K=1,6)
                REWIND(5)
            ELSE
                GO TO 2810
            END IF

2820      REWIND(5)
2800  CONTINUE

        CLOSE (5)
        CALL PITZER (T,0,EX)
        READ(9,*) TINIT
        READ(9,*) IPATH

        IF (IPATH.EQ.1) THEN
            READ(9,*) TFIN
            READ(9,*) DT
        ELSE
            READ(9,*) WFIN
            READ(9,*) DW
        END IF

        T=TINIT
        BALWAT=1000.

1      CONTINUE

        DO 2900 I=IP+1,N
            GIN(I)= PF(T,G0(I,1),G0(I,2),G0(I,3),G0(I,4),G0(I,5),
$          G0(I,6))
2900  CONTINUE

```

```

      IF(IPATH.EQ.1) WRITE(*,*) 'T=', T
      PHI=1.
      CALL PITZER (T,1,EX)

2      CONTINUE

      IF(IPATH.EQ.2) WRITE (*,*) 'AMOUNT OF WATER=', BALWAT
      CALL SOL (ICK,EX,ISOLU,IPATH)
      TC=T-273.15
      WRITE(10,15) TC,T
      IF(IPATH.EQ.2) WRITE(10,16) BALWAT

      IF(ICK.GT.400) GO TO 5

      IF(IOPEN.GT.0) THEN
        WRITE(10,17)

        DO 3000 I=1,IOPEN
          J=JOPEN(I)
          WRITE(10,18) I,NM(NUM(J)),X(J),G(J)
3000      CONTINUE

      END IF

      IF(ISOLU.EQ.0) GO TO 4

C..... START OF OUTPUT .....

      WRITE(10,*) '          AQUEOUS SOLUTION'
      WRITE(10,19) UM, PHI
      WRITE(10,20)

      DO 3100 K=IP1,NL-1
        EX(K)=EX(K)*XW
        G(K)=G(K)-DLOG(XW)
        AC=EXP(ACT(K))
        WRITE(10,21) K-IP,NM(K),X(K),EX(K),EX(K)*AC,AC,G(K)
3100      CONTINUE

      WRITE(10,22) NL-IP,NM(NL),X(NL),DEXP(AH2O),G(NL)
4      WRITE(10,23)

      DO 3200 K=1,IP
        BALL(K)=0.
        BALS(K)=0.
3200      CONTINUE

      DO 3300 I=1,IOPEN
        J=JOPEN(I)

        DO 3301 K=1,IP
          BALS(K)=BALS(K)+X(J)*AIN(K,NUM(J))
3301      CONTINUE

3300      CONTINUE

```

```

DO 3400 I=1,IP

      DO 3401 J=IP1,NL
        BALL(I)=BALL(I)+X(J)*AIN(I,J)
3401    CONTINUE

      WRITE(10,24)NEL(I),AIN(I,N1),BALS(I),BALL(I),BALS(I)+BALL(I)
3400    CONTINUE

5      WRITE(10,25) ICK

C..... FINISH .....
      IF(ICK.GT.400) WRITE (10,26)

      IF (IPATH.EQ.1) THEN
        T=T-DT
        IF(T.GE.TFIN-.001) GO TO 1
      ELSE
        BALWAT=BALWAT-DW
        IF(BALWAT.GE.WFIN) THEN
          AIN(1,N1)=BALWAT/18.0153
          GO TO 2
        END IF
      END IF

      CLOSE(3)

815    STOP
      END

```

C-----C

C-----C

```

      SUBROUTINE CHOICE (NCAT,BK,NMCUR,NUMCUR,IZC)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

C.. SUBROUTINE FOR CHOOSING COMPONENTS FROM DATABASE ACCORDING TO
C.. THE GIVEN INDEPENDENT COMPONENTS

      CHARACTER *40 NM,NMCUR
      DIMENSION BK(10,20),NMCUR(20),NUMCUR(20),IZC(20),B(10)
      COMMON /COMPNT/ NUR(10),IP,NUL(10),IPNUL,KCOMP

3      FORMAT(A23,I2,1X,I2,1X,10F6.2)

      READ(1,*)NCAT
      J=0

      DO 2 I=1,NCAT
c      READ(1,3) NM,NUM,IZ, (B(I1),I1=1, KCOMP)
        READ(1,*) NM,NUM,IZ, (B(I1),I1=1, KCOMP)

        DO 2000 J1=1,IPNUL
          IF (B(NUL(J1)).NE.0.) GO TO 2
2000      CONTINUE

```

```

        IS=0

        DO 3000 J1=1,IP
            IF (DABS(B(NUR(J1))).GT.1D-20) IS=IS+1
3000    CONTINUE

        IF (IS.EQ.0) GO TO 2
        J=J+1

        DO 4 J1=1,IP
            BK(J1,J)=B(NUR(J1))
4        CONTINUE

        NMCUR(J)=NM
        NUMCUR(J)=NUM
        IZC(J)=IZ
2        CONTINUE

        NCAT=J

        RETURN
        END
C-----C
C-----C
        SUBROUTINE INTACT(Z1,Z2,UM,A,PHIPHI,PHIPRI,PHIIJ,THETA)
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)

C    THIS SUBROUTINE CALCULATES THE HIGHER-ORDER ELECTROSTATIC
C    INTERACTION TERMS FOR PITZER EQUATIONS.

        DIMENSION B(0:22), D(0:22), ZA(2), XA(2,2),XJ(2,2),XJPRIM(2,2)
        DIMENSION AKI(0:20), AKII(0:20)

        DATA AKI/1.925154014814667, -0.060076477753119,
$ -0.029779077456514, -0.007299499690937, 0.000388260636404,
$ 0.000636874599598, 0.000036583601823, -0.000045036975204,
$ -0.000004537895710, 0.000002937706971, 0.000000396566462,
$ -0.000000202099617, -0.000000025267769, 0.000000013522610,
$ 0.000000001229405, -0.000000000821969, -0.00000000050847,
$ 0.00000000046333, 0.000000000001943, -0.00000000002563,
$ -0.000000000010991/

        DATA AKII/0.628023320520852, 0.462762985338493,
$ 0.150044637187895, -0.028796057604906, -0.036552745910311,
$ -0.001668087945272, 0.006519840398744, 0.001130378079086,
$ -0.000887171310131,-0.000242107641309, 0.000087294451594,
$ 0.000034682122751,-0.000004583768938, -0.000003548684306,
$ -0.000000250453880, 0.000000216991779, 0.000000080779570,
$ 0.000000004558555,-0.000000006944757, -0.000000002849257,
$ 0.000000000237816/

        B(21)=0.
        B(22)=0.
        D(21)=0.
        D(22)=0.
        ZA(1)=Z1
        ZA(2)=Z2
        SQ=SQRT(UM)

```

```

DO 2000 J=1,2

      DO 2001 I =1,2
        XA(J,I)=6.*ZA(I)*ZA(J)*A*SQ
        X=XA(J,I)
        IF (X.LT.1.) THEN
          ZZ=4.*X**0.2-2.0
          DZ=.8*X**(-.8)

          DO 2002 K=20,0,-1
            B(K)=ZZ*B(K+1)-B(K+2)+AKI(K)
            D(K)=B(K+1)+ZZ*D(K+1)-D(K+2)
2002      CONTINUE

          ELSE
            ZZ=40./9.*X**(-.1)-22./9.
            DZ=-40./90.*X**(-1.1)

            DO 2003 K=20,0,-1
              B(K)=ZZ*B(K+1)-B(K+2)+AKII(K)
              D(K)=B(K+1)+ZZ*D(K+1)-D(K+2)
2003      CONTINUE

          END IF
          XJ(J,I)=.25*X-1.+5*(B(0)-B(2))
          XJPRIM(J,I)=.25+.5*DZ*(D(0)-D(2))
2001      CONTINUE

2000    CONTINUE

    ETHETA=(Z1*Z2/4./UM)*(XJ(1,2)-.5*XJ(1,1)-.5*XJ(2,2))
    ETHPRI=-ETHETA/UM+(Z1*Z2/8./UM**2)*(XA(1,2)*XJPRIM(1,2)-
$    .5*XA(1,1)*XJPRIM(1,1)-.5*XA(2,2)*XJPRIM(2,2))
    PHIPHI=THETA+ETHETA+UM*ETHPRI
    PHIIJ=THETA+ETHETA
    PHIPRI=ETHPRI

    RETURN
    END

C-----C
C-----C

SUBROUTINE PITZER (T,IFLAG,EX)
  IMPLICIT DOUBLE PRECISION (A-H,O-Z)

  DIMENSION A0M(6),NMM(40),NAQ(20),Z(20),EX(20)
  DIMENSION BPRIME(10,10),BPHI(10,10),C(10,10),PHIIJ(10,10),
$  SUMCA(10),SUMCAT(10),SUMAN(10),SUMZ(10),SUMAC(10),ACT(20),
$  PHIPHI(10,10),PHIPRI(10,10),XM(40),IZ(20)
  DIMENSION SUMAA(10),SUMCC(10),SUMK(10)
  DIMENSION BET0(10,10),BET1(10,10),C0(10,10),BET2(10,10),
$  TET(10,10),PSI(10,10,10),B(10,10)
  DIMENSION BET0M(10,10,6),BET1M(10,10,6),COM(10,10,6),
$  BET2M(10,10,6),TETM(10,10,6),PSIM(10,10,10,6)

  COMMON /NUMBER/NCAT,NANI,NNEI,NMM

```

```

COMMON /MATRIX/ ACT,UM,PHI,AH2O,IZ,IP,NMV

IF (IFLAG.EQ.1) GO TO 1
IF (IFLAG.EQ.2) GO TO 2

ICOL=6
NN=NCAT+NANI
NC1=NCAT+1

DO 2000 I=1,NN
    Z(I)=IZ(I+IP)
    NAQ(I)=NMM(I+IP)
2000 CONTINUE

C..... READ TABLES 1 AND 2 .....

OPEN(3, FILE='TABLE1')

C..... READ TABLE 1 .....

DO 3000 I=1,NN
    DO 3001 J=1,NN
        DO 3002 K=1,6
            COM(I,J,K)=0.
            BET2M(I,J,K)=0.
3002 CONTINUE
3001 CONTINUE
3000 CONTINUE

READ(3,*) (A0M(I), I=1,6)

DO 4000 I=1,NCAT
    DO 4001 J=NC1,NN
101 READ(3,FMT='(2I3)') NT1,NT2

    IF(NAQ(I).EQ.NT1.AND.NAQ(J).EQ.NT2) THEN
        READ(3,*) (BET0M(I,J,K),K=1,6)
        READ(3,*) (BET1M(I,J,K),K=1,6)

        IF((NT1.EQ.3 .OR. NT1.EQ.4) .AND. NT2.EQ.12)THEN
            READ(3,*) (BET2M(I,J,K),K=1,6)
            READ(3,*) (COM(I,J,K),K=1,6)
        ELSE
            READ(3,*) (COM(I,J,K),K=1,6)
        END IF

    ELSE

        IF((NT1.EQ.3 .OR. NT1.EQ.4) .AND. NT2.EQ.12)THEN
            READ(3,FMT='(///)',END=910)
        ELSE
            READ(3,FMT='(///)',END=910)
        END IF

```

```

                                GO TO 101

                                END IF
4001      CONTINUE

4000  CONTINUE

      CLOSE(3)

C..... READ TABLE 2 .....

      OPEN(4, FILE='TABLE2')

      DO 5000 I=1,NCAT-1

          DO 5001 J=I+1,NCAT
102      READ(4,*) NT1,NT2,NT3

              IF(NAQ(I).EQ.NT1.AND.NAQ(J).EQ.NT2.AND.NT3.EQ.0) THEN
                  BACKSPACE(4)
                  READ(4,*) NT1,NT2,NT3,(TETM(I,J,K),K=1,6)
                  REWIND(4)
              ELSE
                  GO TO 102
              END IF

5001      CONTINUE

5000  CONTINUE

      DO 6000 I=1,NCAT-1

          DO 6001 J=I+1,NCAT

              DO 6002 IJ = NC1,NN
105      READ(4,*) NT1,NT2,NT3

                  IF(NAQ(I).EQ.NT1
$                  .AND. NAQ(J).EQ.NT2
$                  .AND. NAQ(IJ).EQ.NT3) THEN
                      BACKSPACE(4)
                      READ(4,*)NT1,NT2,NT3,(PSIM(I,J,IJ,K),K=1,6)
                      REWIND(4)
                  ELSE
                      GO TO 105
                  END IF

6002      CONTINUE

6001      CONTINUE

6000  CONTINUE

      DO 7000 I=NC1,NN-1

          DO 7001 J=I+1,NN
103      READ(4,*) NT1,NT2,NT3

```



```

                IF(NAQ(I) .EQ. NT1
$                .AND.  NAQ(J) .EQ. NT2
$                .AND.  NT3.EQ.0) THEN
                BACKSPACE(4)
                READ(4,*) NT1,NT2,NT3,(TETM(I,J,K),K=1,6)
                REWIND(4)
                ELSE
                GO TO 103
            END IF

7001          CONTINUE

7000  CONTINUE

        DO 8000 I=NC1,NN-1

            DO 8001 J=I+1,NN

                DO 8002 IA=1,NCAT
106          READ(4,*) NT1,NT2,NT3

                    IF(NAQ(I) .EQ. NT1
$                    .AND.  NAQ(J) .EQ. NT2
$                    .AND.  NAQ(IA) .EQ. NT3) THEN
                    BACKSPACE(4)
                    READ(4,*)NT1,NT2,NT3,(PSIM(I,J,IA,K),K=1,6)
                    REWIND(4)
                    ELSE
                    GO TO 106
                END IF

8002          CONTINUE

8001          CONTINUE

8000  CONTINUE

104  CLOSE(4)

C.....  END OF READING .....

        RETURN

1      CONTINUE

C..  CALCULATION OF VALUES OF THE PARAMETERES FOR CURRENT TEMPERATURE

        A0= PF(T,A0M(1),A0M(2),A0M(3),A0M(4),A0M(5),A0M(6))

        DO 9000 I=1,NCAT

            DO 9001 J=NC1,NN

                BET0(I,J)= PF(T,BET0M(I,J,1),BET0M(I,J,2),
$BET0M(I,J,3),BET0M(I,J,4),BET0M(I,J,5),BET0M(I,J,6) )
                BET1(I,J)= PF(T,BET1M(I,J,1),BET1M(I,J,2),
$BET1M(I,J,3),BET1M(I,J,4),BET1M(I,J,5),BET1M(I,J,6) )
                BET2(I,J)= PF(T,BET2M(I,J,1),BET2M(I,J,2),
$BET2M(I,J,3),BET2M(I,J,4),BET2M(I,J,5),BET2M(I,J,6) )
                C0(I,J)= PF(T,C0M(I,J,1),C0M(I,J,2),C0M(I,J,3),

```

```

$          COM(I,J,4),COM(I,J,5),COM(I,J,6) )
9001      CONTINUE

9000  CONTINUE

      DO 10000 I=1,NN-1

          DO 10001 J=I+1,NN
              TET(I,J)= PF(T,TETM(I,J,1),TETM(I,J,2),TETM(I,J,3),
$                  TETM(I,J,4),TETM(I,J,5),TETM(I,J,6))

              DO 10002 IA=1,NN
                  PSI(I,J,IA)=PF(T,PSIM(I,J,IA,1),PSIM(I,J,IA,2),
$                      PSIM(I,J,IA,3),PSIM(I,J,IA,4),
$                      PSIM(I,J,IA,5),PSIM(I,J,IA,6))
10002      CONTINUE

10001  CONTINUE

10000 CONTINUE

      RETURN

2      CONTINUE

      DO 11000 I = 1, 10
          SUMCA(I) = 0.
          SUMCAT(I) = 0.
          SUMAN(I) = 0.
          SUMZ(I) = 0.
          SUMAC(I) = 0.
          SUMAA(I) = 0.
          SUMCC(I) = 0.
          SUMK(I) = 0.
11000 CONTINUE

      SMX=0.
      ZZ=0.
      UM=0.

      DO 12000 I=IP+1,NMV
          J=I-IP
          XM(J)=EX(I)*55.50837
          ZZ=ZZ+XM(J)*DABS(Z(J))
          UM=UM+XM(J)*Z(J)**2
          SMX=SMX+XM(J)
12000 CONTINUE

      UM=UM/2.
      SQ=SQRT(UM)
      ALPHA=2.*SQ
      ALPHA1=1.4*SQ
      ALPHA2=12.*SQ
      G1=2.*(1.-(1.+ALPHA1)*EXP(-ALPHA1))/ALPHA1**2
      G2=2.*(1.-(1.+ALPHA2)*EXP(-ALPHA2))/ALPHA2**2
      GPRI1=-2.*(1.-(1.+ALPHA1+ALPHA1**2/2.)*EXP(-ALPHA1))/ALPHA1**2
      GPRI2=-2.*(1.-(1.+ALPHA2+ALPHA2**2/2.)*EXP(-ALPHA2))/ALPHA2**2
      G=2.*(1-(1+ALPHA)*EXP(-ALPHA))/ALPHA**2
      GPRIME=-2.*(1.-(1.+ALPHA+ALPHA**2/2.)*EXP(-ALPHA))/ALPHA**2

```

```

DO 13000 J=1,NCAT

      DO 13001 I=NC1,NN

            IF (Z(J)*ABS(Z(I)).EQ.4) THEN
                  BPHI(J,I)=BET0(J,I)+BET1(J,I)*EXP(-ALPHA1)+
$                  BET2(J,I)*EXP(-ALPHA2)
                  B(J,I)=BET0(J,I)+BET1(J,I)*G1+BET2(J,I)*G2
                  BPRIME(J,I)=BET1(J,I)*GPRI1/UM+BET2(J,I)*GPRI2/UM
            ELSE
                  BPHI(J,I)=BET0(J,I)+BET1(J,I)*EXP(-ALPHA)
                  B(J,I)=BET0(J,I)+BET1(J,I)*G
                  BPRIME(J,I)=BET1(J,I)*GPRIME/UM
            END IF

            C(J,I)=C0(J,I)/2./SQRT(Z(J)*DABS(Z(I)))
13001      CONTINUE

13000 CONTINUE

      DO 14000 J=1,NCAT-1

            DO 14001 I=J+1,NCAT
                  CALL INTACT(Z(J),Z(I),UM,A0,PHIPHI(J,I),
$                  PHIPRI(J,I),PHIIJ(J,I),TET(J,I))
14001      CONTINUE

14000 CONTINUE

      DO 15000 J=NC1,NN-1

            DO 15001 I=J+1,NN
                  CALL INTACT(Z(J),Z(I),UM,A0,PHIPHI(J,I),
$                  PHIPRI(J,I),PHIIJ(J,I),TET(J,I))
15001      CONTINUE

15000 CONTINUE

C..   CALCULATION OF SUMMATION TERMS FOR F AND PHI.

SCATON=0.
SUBSUM=0.
SANON=0.
SUMCAF=0.
SUMANF=0.

DO 16000 J=1,NCAT-1

      DO 16001 J1=J+1,NCAT

            DO 16002 I=NC1,NN
                  SUBSUM=SUBSUM+PSI(J,J1,I)*XM(I)
16002      CONTINUE

            SCATON=SCATON+(SUBSUM+PHIPHI(J,J1))*XM(J)*XM(J1)
            SUMCAF=SUMCAF+PHIPRI(J,J1)*XM(J)*XM(J1)
            SUBSUM=0.
16001      CONTINUE

```

```

16000 CONTINUE

      SUBSUM=0.

      DO 17000 J=NC1,NN-1

          DO 17001 J1=J+1,NN

              DO 17002 I=1,NCAT
                  SUBSUM=SUBSUM+PSI(J,J1,I)*XM(I)
17002          CONTINUE

                  SANON=SANON+(SUBSUM+PHIPHI(J,J1))*XM(J)*XM(J1)
                  SUMANF=SUMANF+PHIPRI(J,J1)*XM(J)*XM(J1)
                  SUBSUM=0.
17001          CONTINUE
      CONTINUE

17000 CONTINUE

      SUMB=0.
      SUMPFI=0.

      DO 18000 J=1,NCAT

          DO 18001 I=NC1,NN
              SUMB=SUMB+XM(J)*XM(I)*BPRIME(J,I)
              SUMPFI=SUMPFI+XM(J)*XM(I)*(BPHI(J,I)+ZZ*C(J,I))
18001          CONTINUE
      CONTINUE

18000 CONTINUE

      F=-A0*(SQ/(1.+1.2*SQ)+2.*DLOG(1.+1.2*SQ)/1.2)+
$      SUMB+SUMCAF+SUMANF
      PHI=1.+2./SMX*(-A0*UM**1.5/(1.+1.2*SQ)+SUMPFI+SCATON+SANON)
      AH2O=-PHI*SMX/55.50837

C..... CALCULATION OF TERMS FOR ACTIVITY COEFFICIENTS(GAMMA).....

      SUM=0.

      DO 19000 J=1, NCAT-1

          DO 19001 J1=J+1, NCAT

              DO 19002 I=NC1,NN
                  PSI(J1,J,I)=PSI(J,J1,I)
19002          CONTINUE

                  PHIIJ(J1,J)=PHIIJ(J,J1)
19001          CONTINUE
      CONTINUE

19000 CONTINUE

      DO 20000 I=NC1, NN-1

          DO 20001 I1=I+1, NN

              DO 20002 J=1, NCAT
                  PSI(I1,I,J)=PSI(I,I1,J)

```

```

20002          CONTINUE

                PHIIJ(I1,I)=PHIIJ(I,I1)
20001          CONTINUE

20000 CONTINUE

        DO 21000 J=1, NCAT

                DO 21001 I=NC1,NN
                        SUMCA(J)=SUMCA(J)+XM(I)*(2.*(B(J,I))+ZZ*C(J,I))
21001          CONTINUE

21000 CONTINUE

        DO 22000 J=1, NCAT

                DO 10 J1=1,NCAT
                        IF (J.EQ.J1) GO TO 10

                        DO 22001 I=NC1, NN
                                SUM=SUM+XM(I)*PSI(J,J1,I)
22001          CONTINUE

                                SUMCAT(J)=SUMCAT(J)+XM(J1)*(SUM+2.*PHIIJ(J,J1))
                                SUM=0.
10          CONTINUE

22000 CONTINUE

        DO 23000 J=1, NCAT

                DO 23001 J1=NC1,NN-1

                        DO 23002 I=J1+1, NN
                                SUMAN(J)=SUMAN(J)+XM(J1)*XM(I)*PSI(J1,I,J)
23002          CONTINUE

23001          CONTINUE

23000 CONTINUE

                SUM=0.

        DO 24000 J=1,NCAT

                DO 24001 I=NC1,NN
                        SUM=SUM+XM(J)*XM(I)*C(J,I)
24001          CONTINUE

24000 CONTINUE

        DO 25000 J=1,NCAT
                SUMZ(J)=SUM*DABS(Z(J))
25000 CONTINUE

        DO 26000 J=1,NCAT
                ACT(J+IP)=Z(J)**2*F+SUMCA(J)+SUMCAT(J)+SUMAN(J)+SUMZ(J)

```

```

26000 CONTINUE

      SUM=0.

      DO 27000 I=NC1, NN

          DO 27001 J=1, NCAT
              SUMAC(I)=SUMAC(I)+XM(J)*(2.*(B(J,I))+ZZ*C(J,I))
27001          CONTINUE

27000 CONTINUE

      DO 28000 I=NC1, NN

          DO 20 I1=NC1,NN
              IF (I.EQ.I1) GO TO 20

              DO 28001 J=1, NCAT
                  SUM=SUM+XM(J)*PSI(I,I1,J)
28001          CONTINUE

              SUMAA(I)=SUMAA(I)+XM(I1)*(SUM+2.*PHIJ(I,I1))
              SUM=0.
20          CONTINUE

28000 CONTINUE

      DO 29000 I=NC1, NN

          DO 29001 J=1,NCAT-1

              DO 29002 J1=J+1, NCAT
                  SUMCC(I)=SUMCC(I)+XM(J)*XM(J1)*PSI(J,J1,I)
29002          CONTINUE

29001          CONTINUE

29000 CONTINUE

      SUM=0.

      DO 30000 J=1,NCAT

          DO 30001 I=NC1,NN
              SUM=SUM+XM(J)*XM(I)*C(J,I)
30001          CONTINUE

30000 CONTINUE

      DO 31000 I=NC1,NN
          SUMK(I)=SUM*ABS(Z(I))
31000 CONTINUE

      DO 32000 J=NC1,NN
          ACT(J+IP)=Z(J)**2*F+SUMAC(J)+SUMAA(J)+SUMCC(J)+SUMK(J)
32000 CONTINUE

      DO 33000 I=1,NMV-IP
          IF (ACT(I+IP).GT.ALOG(10000.)) ACT(I+IP)=ALOG(10000.)

```

33000 CONTINUE

RETURN

910 WRITE(*,1910) NAQ(I),NAQ(J)

1910 FORMAT(2(1X,I3),/,',', '!! TABLE 1 IS NOT COMPLETE !!')/

\$' !! OR WRONG COMPONENT WAS ENTERED !!')

STOP

END

C-----C

C-----C

SUBROUTINE SIMPL (IP1,M,IOP1,IP,N1,K,IR)

IMPLICIT DOUBLE PRECISION (A-H,O-Z)

COMMON /MASS1/ A,G,NUM,X(40)

COMMON /ENTER/ ISIM,ISIMIN

DIMENSION A(10,40),G(40),NUM(40)

C IP1 - BEGINING J, M - FINISH J, IOP1- BEGINING I, IP -FINISH I

C.....LOOKING FOR A NEW PHASE FOR INCLUDING INTO BASIS.....

ISIM=0

1111 DEL=0.D0

IK=0

K=0

DO 1001 J=IP1,M

P=0.

DO 1002 I=1,IP

P=P+G(I)*A(I,J)

1002 CONTINUE

DELTA=G(J)-P

IF(DELTA.GE.DEL) GO TO 1001

DEL=DELTA

K=J

ISIM=1

1001 CONTINUE

IF(K.EQ.0) GO TO 1000

C.....LOOKING FOR PLACE IN THE BASIS TO BE SUBSTITUTED.....

BMIN=1D+20

DO 1005 I=IOP1,IP

IF(A(I,K).LE.0.) GO TO 1005

BTEK=A(I,N1)/A(I,K)

IF(BTEK.GE.BMIN) GO TO 1005

BMIN=BTEK

IR=I

1005 CONTINUE

DO 1007 I=1,IP

A(I,IR)=A(I,K)

1007 CONTINUE

```

      NUM(IR)=NUM(K)
      G(IR)=G(K)
      CALL GG1(IR,IP+1,IP,N1)

C..      IF(ISIMIN.EQ.0) GO TO 1111

1000  CONTINUE

      RETURN
      END

C-----C
C-----C

      SUBROUTINE GG1(ICOL,IP1,IPS,N1)
      DOUBLE PRECISION A,G,X
      DIMENSION A(10,40),G(40),NUM(40)
      COMMON /MASS1/ A,G,NUM,X(40)

      DO 1 I=IP1,N1
        A(ICOL,I)=A(ICOL,I)/A(ICOL,ICOL)
1      CONTINUE

      A(ICOL,ICOL)=1.

      DO 2 J=1,IPS
        IF(J.EQ.ICOL) GO TO 2

        DO 3 I=IP1,N1
          A(J,I)=A(J,I)-A(ICOL,I)*A(J,ICOL)
3      CONTINUE

        A(J,ICOL)=0.
2      CONTINUE

      RETURN
      END

C-----C
C-----C

      SUBROUTINE GG(INI,NP,NP1)
      DOUBLE PRECISION P1,AM,AT
      COMMON/COMGG/ P1(20,21)

      DO 201 I=INI,NP
        AM=P1(I,I)
        I1=I
        IJ=I+1

        DO 202 K=IJ,NP
          IF(ABS(AM).GE.DABS(P1(K,I))) GO TO 202
          AM=P1(K,I)
          I1=K
202      CONTINUE

```



```

      DO 203 K=I,NP1
        AT=P1(I1,K)
        P1(I1,K)=P1(I,K)
        P1(I,K)=AT/AM
203    CONTINUE

      DO 241 J=INI,NP
        AM=P1(J,I)
        IF(J.EQ.I) GO TO 241

        DO 204 K=I,NP1
          P1(J,K)=P1(J,K)-P1(I,K)*AM
204    CONTINUE

241    CONTINUE

201    CONTINUE

      RETURN
      END

C-----C
C-----C

C..  THE ROUTINE FOR CALCULATION OF CHEMICAL EQUILIBRIA
C..  IN WATER-SALT SYSTEMS AT THE TEMPERATURE RANGE 25- -60 C.

      DOUBLE PRECISION FUNCTION PF(T,A1,A2,A6,A9,A3,A4)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)

      PF=A1+A2*T+A6*T**2+A9*T**3+A3/T+A4*LOG(T)

      RETURN
      END

C-----C
C-----C

      SUBROUTINE SOL (ICK,EX,ISOLU,IPATH)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      CHARACTER *40 NM

      DIMENSION G(40),X(40),EX(20),EXTMP(20),A(10,40),ACT(20),NM(40)
      DIMENSION JOPEN(10),JCLOS(10),SU(20),P1(20,21),AIN(10,40)
      DIMENSION NUMIN(40),NUM(40),GIN(40),ACTT(20),IZ(20)
      COMMON /NUMBER/NCAT,NANI,NNEI,NMM(40)
      COMMON /MASS1/ A,G,NUM,X
      COMMON /MASS2/ NM
      COMMON /INTGER/ NL,N,N1
      COMMON /MATRIX/ ACT,UM,PHI,AH2O,IZ,IP,NMV
      COMMON /AIN/ AIN,GIN,T,NUMIN,IV
      COMMON /COMGG/ P1
      COMMON /ENTER/ ISIM,ISIMIN
      COMMON /OUTPUT/JOPEN,IOPEN

      DO 1000 I = 1, 20
        ACT(I) = 0.D0
1000 CONTINUE

```

```

ISIMIN=1
ISOLU=1
ICK=0
ICE=0
XW=55.50837

DO 2000 I = 1, 10

    DO 2001 J = 1, 40
        A(I,J)=AIN(I,J)
2001    CONTINUE

2000 CONTINUE

    DO 2100 I = 1, 40
        NUM(I)=NUMIN(I)
        G(I)=GIN(I)
2100 CONTINUE

    EPS=.001
    IP1=IP+1
    NUM(N1)=N1
    NM(N1)='DEL'
    PHI=1.
    IP1=IP+1
    NL1=NL+1
    L=NL-IP
    NMV=NL-1
    N1=N+1

    DO 2200 I=IP1,NMV
        G(I)=G(I)+DLOG(XW)
2200 CONTINUE

    DO 2300 I=1,IP

        DO 2301 J=IP1,NL
            IF (NMM(I).EQ.NMM(J)) THEN
                NUM(I)=J
                NUMIN(I)=J
                G(I)=G(J)
            END IF
2301 CONTINUE

            IF (NMM(I).EQ.30) THEN
                IVV=I
                X(IV)=A(I,N1)
            END IF
2300 CONTINUE

    SUM=0.D0

    DO 2400 I=1,IP
        NI=NUM(I)
        IF (I.NE.IVV) THEN
            X(NI)=A(I,N1)
            EX(NI)=A(I,N1)/X(IV)
            G(I)=G(NUM(I))+DLOG(EX(NUM(I)))

```

```

                END IF
2400  CONTINUE

        CALL PITZER (T,2,EX)
        G(IVV)=AH2O

        DO 2500 I=1,IP
            NI=NUM(I)
            IF (I.NE.IVV) THEN
                G(I)=G(NI)+DLOG(EX(NI))+ACT(NI)
            END IF
2500  CONTINUE

        DO 2600 J=IP1,NMV
            SU(J)=-G(J)

                DO 2601 K=1,IP
                    SU(J)=SU(J)+A(K,J)*G(K)
2601  CONTINUE

                EX(J)=DEXP(SU(J)-ACT(J))
                X(J)=X(IV)*EX(J)
2600  CONTINUE

        CALL SIMPL (NL1,NL1,1,IP,N1,IK,IR)
        DELL=1.
75  CONTINUE
76  ICLOS=0
    IOPEN=0

        DO 2700 I=1,IP
            IF (NUM(I).GE.NL1.AND.NUM(I).LE.N) THEN
                IOPEN=IOPEN+1
                JOPEN(IOPEN)=I
            ELSE
                ICLOS=ICLOS+1
                JCLOS(ICLOS)=I
            END IF
2700  CONTINUE

        IF (IOPEN.GT.0) THEN
            IF (IOPEN.EQ.1.AND.NMM(NUM(JOPEN(1))).EQ.31) THEN
                ICE=1
                SU(IV)=-G(IV)

                DO 2800 K=1,IP
                    SU(IV)=SU(IV)+A(K,IV)*G(K)
2800  CONTINUE

                XMN=1.

                DO 2900 K=1,10
                    XMN=XMN*1.5

                    DO 2901 J=IP1,NMV
                        EXTMP(J)=EX(J)*XMN
                        SUMS=SUM*XMN
2901  CONTINUE

```

```

        CALL PITZER (T,2,EXTMP)
        IF (AH2O.LE.SU(IV)) GO TO 10
2900      CONTINUE

10        X(IV)=X(IV)/XMN*1.2

        DO 3000 I=1,ICLOS
            J=JCLOS(I)
            NJ=NUM(J)
            EX(NJ)=EXTMP(NJ)/1.2
            G(J)=G(NJ)+DLOG(EX(NJ))+ACT(NJ)
3000      CONTINUE
        ELSE

            DO 3200 I=1,IP

                DO 3201 J=IP1, NL
                    IF (NUMIN(I).EQ.J) THEN
                        X(J)=X(J)-.95*A(IR,N1)*AIN(I,IK)
                        IF (X(J).LE.0.) X(J)=1.E-5
                    END IF
3201      CONTINUE

3200      CONTINUE

            IF (ICE.EQ.1) THEN
                SU(IV)=-G(IV)

                DO 3300 K=1,IP
                    SU(IV)=SU(IV)+A(K,IV)*G(K)
3300      CONTINUE

                III=0
            END IF

11        DO 3400 J=IP1,NMV
            EX(J)=X(J)/X(IV)
3400      CONTINUE

            III=III+1

            CALL PITZER (T,2,EX)

            IF (ICE.EQ.1.AND.IPATH.EQ.1) THEN
                IF (AH2O.GT.SU(IV)) THEN
                    X(IV)=X(IV)/1.2
                    GO TO 11
                ELSE
                    IF (III.GT.1) X(IV)=X(IV)*1.1
                END IF
            END IF

            DO 3500 I=1,ICLOS
                J=JCLOS(I)
                NJ=NUM(J)
                IF (NJ.LT.NL) G(J)=G(NJ)+DLOG(EX(NJ))+ACT(NJ)
3500      CONTINUE

```

```

        DO 3600 J=IP1,NMV
            SU(J)=-G(J)

            DO 3601 K=1,IP
                SU(J)=SU(J)+A(K,J)*G(K)
3601        CONTINUE

            IF(J.NE.IV) EX(J)=DEXP(SU(J)-ACT(J))
            X(J)=EX(J)*X(IV)
3600        CONTINUE

        END IF
    END IF

    IN=ICLOS+L
    IN1=IN+1

5        IF(ICK.GT.400) RETURN
    ICK=ICK+1

C..    EXAMINE FOR AQUEOUS SOLUTION PRESENCE.....

    IF(ICLOS.LE.1) THEN
        ISOLU=0

        DO 3700 I=IP1,NL
            X(I)=0.
3700        CONTINUE

        GO TO 400
    END IF

    ISL=0

    DO 3800 J=IP1,NL
        SU(J)=G(J)

        DO 3801 K=1,IP
            SU(J)=SU(J)-A(K,J)*G(K)
3801        CONTINUE

        EX(J)=X(J)/X(IV)
        ACTT(J)=ACT(J)
        IF(IS.EQ.1) ACTT(J)=0.
3800        CONTINUE

    IS=IS+1
    PHITT=PHI

    CALL PITZER (T,2,EX)

    PHI=(PHITT+PHI)/2
    SUM=0.

    DO 3900 J=IP1,NMV
        ACT(J)=(ACT(J)+ACTT(J))/2.
        JI=J-IP
        P1(JI,JI)=1./X(J)

```

```

      P1(JI,IN1)=DLOG(X(J))+ACT(J)-DLOG(X(IV))+SU(J)
      P1(JI,L)=-1./X(IV)
      P1(L,JI)=-1./X(IV)*PHI
      SUM=SUM+EX(J)
3900  CONTINUE

      P1(L,L)=SUM*PHI/X(IV)
      P1(L,IN1)=SU(NL)-SUM*PHI

      DO 4000 I=1,ICLOS
        K=L+I
        II=JCLOS(I)
        P1(K,K)=0.
        P1(K,IN1)=A(II,N1)

        DO 4001 J=IP1,NL
          P1(K,IN1)=P1(K,IN1)-A(II,J)*X(J)
          IJ=J-IP
          P1(K,IJ)=-A(II,J)
          P1(IJ,K)=-A(II,J)
4001  CONTINUE

4000  CONTINUE

      CALL GG (1,IN,IN1)

      DO 4100 I=1,ICLOS
        K=L+I
        II=JCLOS(I)
        YLI=G(II)-P1(K,IN1)/DELL
        XBS=DABS(YLI-G(II))
        IF(XBS.LE.EPS) ISL=ISL+1
        G(II)=YLI
4100  CONTINUE

      DO 4200 I=1,L
        J=I+IP
        YLI=X(J)-P1(I,IN1)/DELL
        IF(YLI.LE.1.D-10) YLI=1.D-6
        XBS=DABS((YLI-X(J))/X(J))
        IF(XBS.LE.EPS) ISL=ISL+1
        X(J)=YLI
4200  CONTINUE

      IF(X(NL).LE.1.D-3) X(NL)=1.D-3
      IF(ISL.LT.IN) GO TO 5

C..    DELETING OF FIXED COMPOSITION PHASES

400    XX=0.
      IMIN=0

      DO 3100 I=1,IOPEN
        JO=JOPEN(I)
        SUM=0.

        DO 3101 J=IP1,NL
          SUM=SUM+A(JO,J)*X(J)
3101  CONTINUE

```

```

        X(JO)=A(JO,N1)-SUM
        IF(X(JO).GT.XX)GO TO 3100
        XX=X(JO)
        IMIN=JO
3100  CONTINUE

        IF (ISOLU.EQ.0) RETURN
        IF (IMIN.EQ.0) GO TO 1025
        NUM(IMIN)=N1
        GO TO 76

1025  CALL SIMPL(NL1,N,1,IP,N1,IK,IR)

        IF(ISIM.EQ.1) THEN
            DELL=DELL+.5
            GO TO 75
        END IF

7000  CONTINUE

        RETURN
        END

```

APPENDIX B: DATA FILES FOR PROGRAM FREZCHEM2

File DATABASE

9 - number of independent components

H2O 0 30 - the name of the independent component, its charge and its

Na+ 1 1 code (the codes in this model are the same as

K+ 1 2 in the FREZCHEM model)

Ca2+ 2 3

Mg2+ 2 4

Cl- -1 11

SO4-2 -2 12

CO3-2 -2 15

H+ 1 6

5 -number of cations

			H2O	Na+	K+	Ca2+	Mg2+	Cl-	SO4-2	CO3-2	H+
Na+(aq)	1	1	.00	1.00	.00	.00	.00	.00	.00	.00	.00
K+(aq)	2	1	.00	.00	1.00	.00	.00	.00	.00	.00	.00
Ca2+(aq)	3	2	.00	.00	.00	1.00	.00	.00	.00	.00	.00
Mg2+(aq)	4	2	.00	.00	.00	.00	1.00	.00	.00	.00	.00
H+(aq)	5	1	.00	.00	.00	.00	.00	.00	.00	.00	1.00

5 -number of anions

Cl- (aq)	11	-1	.00	.00	.00	.00	.00	1.00	.00	.00	.00
SO4-2 (aq)	12	-2	.00	.00	.00	.00	.00	.00	1.00	.00	.00
OH- (aq)	13	-1	1.00	.00	.00	.00	.00	.00	.00	.00	-1.00
HCO3- (aq)	14	-1	.00	.00	.00	.00	.00	.00	.00	1.00	1.00
CO3-2 (aq)	15	-2	.00	.00	.00	.00	.00	.00	.00	1.00	.00

4 -number of neutral species

CO2(aq)	21	0	-1.00	.00	.00	.00	.00	.00	.00	1.00	2.00
CaSO4(aq)	22	0	.00	.00	.00	1.00	.00	.00	1.00	.00	.00
MgSO4(aq)	23	0	.00	.00	.00	.00	1.00	.00	1.00	.00	.00
H2O(l)	30	0	1.00	.00	.00	.00	.00	.00	.00	.00	.00

16 - number of solid phases

			H2O	Na+	K+	Ca2+	Mg2+	Cl-	SO4-2	CO3-2	H+
H2O(cr, I)	31	0	1.00	.00	.00	.00	.00	.00	.00	.00	0.00
NaCl*2H2O(cr)	32	0	2.00	1.00	.00	.00	.00	1.00	.00	.00	0.00
NaCl(cr)	33	0	.00	1.00	.00	.00	.00	1.00	.00	.00	.00
KCl(cr)	34	0	.00	.00	1.00	.00	.00	1.00	.00	.00	.00
CaCl2*6H2O(cr)	35	0	6.00	.00	.00	1.00	.00	2.00	.00	.00	.00
MgCl2*6H2O(cr)	36	0	6.00	.00	.00	.00	1.00	2.00	.00	.00	.00
MgCl2*8H2O(cr)	37	0	8.00	.00	.00	.00	1.00	2.00	.00	.00	.00
MgCl2*12H2O(cr)	38	0	12.00	.00	.00	.00	1.00	2.00	.00	.00	.00
KMgCl3*6H2O(cr)	39	0	6.00	.00	1.00	.00	1.00	3.00	.00	.00	.00
CaCl2*2MgCl2*12H2O(cr)	40	0	12.00	.00	.00	1.00	2.00	6.00	.00	.00	.00
Na2SO4*10H2O(cr)	41	0	10.00	2.00	.00	.00	.00	.00	1.00	.00	.00
Na2SO4(cr)	42	0	.00	2.00	.00	.00	.00	.00	1.00	.00	.00
MgSO4*6H2O(cr)	43	0	6.00	.00	.00	.00	1.00	.00	1.00	.00	.00
MgSO4*7H2O(cr)	44	0	7.00	.00	.00	.00	1.00	.00	1.00	.00	.00
K2SO4(cr)	45	0	.00	.00	2.00	.00	.00	.00	1.00	.00	.00
MgSO4*K2SO4*6H2O(cr)	46	0	6.00	.00	2.00	.00	1.00	.00	2.00	.00	.00

.

File TABLE1

8.66836498d+01	8.48795942d-02	-8.88785150d-05	4.88096393d-08	-1.32731477d+03	-1.76460172d+01
1 11					
7.87239712d+00	-8.3864096d-03	1.44137774d-05	-8.7820301d-09	-4.96920671d+02	-8.20972560d-01
8.66915291d+02	6.06166931d-01	-4.80489210d-04	1.88503857d-07	-1.70460145d+04	-1.67171296d+02
1.70761824d+00	2.32970177d-03	-2.46665619d-06	1.21543380d-09	-1.35583596d+00	-3.87767714d-01
1 12					
-3.32486330d+03	-2.9297353d+00	2.8024367d-03	-1.316883d-06	5.53958527d+04	6.66660369d+02
-3.57406160d+03	-3.0011206d+00	2.73660950d-03	-1.219171d-06	6.09716482d+04	7.11613120d+02
3.68520478d+02	3.16243995d-01	-2.95372760d-04	1.35491104d-07	-6.22607913d+03	-7.35844094d+01
2 11					
2.65718766d+01	9.92715099d-03	-3.62323330d-06	-6.28427180d-11	-7.55707220d+02	-4.67300770d+00
1.69742977d+03	1.22270943d+00	-9.99044490d-04	4.04786721d-07	-3.28684422d+04	-3.28813848d+02
-3.27571680d+00	-1.27222054d-03	4.71374283d-07	1.1162507d-11	9.07747666d+01	5.80513562d-01
2 12					
4.07908797d+01	8.26906675d-03	0.	0.	-1.41842998d+03	-6.74728848d+00
-1.31669651d+01	2.35793239d-02	0.	0.	2.06712594d+03	0.
-1.88d-02	0.	0.	0.	0.	0.
3 11					
-5.62764702d+01	-3.00771997d-02	1.05630400d-05	3.3331626d-09	1.11730349d+03	1.06664743d+01
3.4787d+00	-1.5417d-02	3.1791d-05	0.	0.	0.
2.64231655d+01	2.46922993d-02	-2.48298510d-05	1.22421864d-08	-4.18098427d+02	-5.35350322d+00
3 12					
1.50d-01	0.	0.	0.	0.	0.
3.00d+00	0.	0.	0.	0.	0.
-1.29399287d+02	4.00431027d-01	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.
4 11					
3.13852913d+02	2.61769099d-01	-2.46268460d-04	1.15764787d-07	-5.53133381d+03	-6.21616862d+01
-3.18432525d+04	-2.86710358d+01	2.78892838d-02	-1.3279705d-05	5.24032958d+05	6.40770396d+03
5.9532d-02	-2.49949d-04	2.41831d-07	0.	0.	0.
4 12					
5.40007849d+03	4.90576884d+00	-4.80489750d-03	2.31126994d-06	-8.80664146d+04	-1.08839565d+03
2.78730869d+00	4.30077440d-03	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.
-5.88623653d+02	-5.05522880d-01	4.8277657d-04	-2.3029838d-07	1.02002016d+04	1.17303808d+02

File TABLE2

1	2	0	-1.82266741d+01	-3.69038470d-03	0.	0.	6.12415011d+02	
3.02994981d+00								
1	2	11	6.48108127d+00	1.46803468d-03	0.	0.	-2.04354019d+02	-
1.09448043d+00								
1	2	12	3.48120d-02	0.	0.	0.	-8.21660d+00	0.
1	3	0	5.0d-02	0.	0.	0.	0.	0.
1	3	11	-7.63980d+00	-1.2990d-02	1.1060d-05	0.	0.	
1.8475d+00								
1	3	12	-1.20d-02	0.	0.	0.	0.	0.
1	4	0	7.0d-02	0.	0.	0.	0.	0.
1	4	11	-3.109870d-02	5.4464780d-05	0.	0.	1.99404210d+00	0.
1	4	12	1.1750520d-01	0.	0.	0.	-4.198620d+01	0.
2	3	0	2.365710d+00	-4.540d-03	0.	0.	-2.84940d+02	0.
2	3	11	-5.930d-02	2.54280d-04	0.	0.	-1.34390d+01	0.
2	3	12	0.	0.	0.	0.	0.	0.
2	4	0	1.1670d-01	0.	0.	0.	0.	0.
2	4	11	5.0362230d-02	-8.750820d-06	0.	0.	-2.899090d+01	0.
2	4	12	-1.3679157d+00	4.24016653d-03	0.	0.	0.	0.
3	4	0	5.31274136d+00	-6.3424248d-03	0.	0.	-9.83113847d+02	0.
3	4	11	4.15790220d+01	1.30377312d-02	0.	0.	-9.81658526d+02	
-7.4061986d+00								
3	4	12	0.	0.	0.	0.	0.	0.
11	12	0	7.00d-02	0.	0.	0.	0.	0.
11	12	1	4.0209775d+00	1.1286005d-03	0.	0.	-1.01169260d+02	-
7.060798d-01								
11	12	2	-2.124815d-01	2.8469833d-04	0.	0.	3.75619614d+01	0.
11	12	3	-1.80d-02	0.	0.	0.	0.	0.
11	12	4	-1.839158d-01	1.429444d-04	0.	0.	3.263d+01	0.

File TABLE3

22	-1.47477745d+01	0.	0.	0.	0.	3.26409496d+03	0.	CaSO4(aq)
23	3.65283115d+02	-7.1578257d-01	0.	0.	0.	-4.48753391d+04	0.	MgSO4(aq)
31	7.875060393d+03	1.169118490d+01	-1.7183789d-02	1.24395543d-05	0.	-9.3314790d+04	-1.7287461d+03	ice
32	-1.2222551d+04	-9.8806459d+00	8.46685083d-03	-3.4459117d-06	0.	2.09823965d+05	2.42328528d+03	hydrohilite
33	9.14839001d+03	8.22348745d+00	-8.1288759d-03	3.95552403d-06	0.	-1.54040868d+05	-1.83624247d+03	halite
34	-1.62917341d+03	-1.51940390d+00	1.45249679d-03	-6.9427505d-07	0.	2.26012743d+04	3.33075506d+02	sylvite
35	1.42290062d+05	1.61973105d+02	-1.95332071d-01	1.17636119d-04	0.	-2.04059847d+06	-2.97464810d+04	antarcticite
36	7.52225099d+02	1.17584653d-01	0.	0.	0.	-2.43223909d+04	-1.21990076d+02	bischofite
37	2.27801976d+03	6.49361616d-01	0.	0.	0.	-6.23075123d+04	-3.95438891d+02	MGCL8
38	2.55008896d+05	2.44532240d+02	-2.48807876d-01	1.22425236d-04	0.	-4.02988342d+06	-5.18668604d+04	MGCL12
39	-4.45702171d+01	2.32023790d-01	-7.14935692d-04	5.32658215d-07	0.	-4.24817923d+03	8.59110245d+00	carnallite
40	8.03777918d+01	-1.388069d-01	0.	0.	0.	0.	0.	tachyhydrite
41	8.44728050d+04	7.68443387d+01	-7.4825816d-02	3.51806085d-05	0.	-1.3881852d+06	-1.7026778d+04	mirabilite
42	-3.9635632d+03	-5.8114490d+00	7.59799462d-03	-4.6571737d-06	0.	3.93454893d+04	8.79598423d+02	thenardite
43	2.39166461d+01	-9.3680744d-02	0.	0.	0.	0.	0.	hexahydrite
44	2.02443627d+00	-1.8051586d-02	0.	0.	0.	0.	0.	epsomite
45	-3.53346251d+04	-2.76809991d+01	2.39856021d-02	-1.0201845d-05	0.	6.37561988d+05	6.94737787d+03	arcanite
46	-1.75203509d+01	2.793713d-02	0.	0.	0.	0.	0.	picromerite

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestion for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE October 1997		3. REPORT TYPE AND DATES COVERED	
4. TITLE AND SUBTITLE FREZCHEM2: A Chemical Thermodynamic Model for Electrolyte Solutions at Subzero Temperatures				5. FUNDING NUMBERS	
6. AUTHORS Mikhail V. Mironenko, Steven A. Grant, Giles M. Marion, and Ronald E. Farren					
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) U.S. Army Cold Regions Research and Engineering Laboratory 72 Lyme Road Hanover, New Hampshire 03755-1290				8. PERFORMING ORGANIZATION REPORT NUMBER CRREL Report 97-5	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Office of the Chief of Engineers U.S. Army Research, Development, Washington, D.C. 20314-1000 and Standardization Group (UK) 223 Old Marylebone Road London NW1 5TH, United Kingdom				10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES For conversion of SI units to non-SI units of measurement consult <i>Standard Practice for Use of the International System of Units (SI)</i> , ASTM Standard E380-93, published by the American Society for Testing and Materials, 1916 Race St., Philadelphia, Pa. 19103.					
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited. Available from NTIS, Springfield, Virginia 22161.				12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) This report documents a Fortran version of a chemical thermodynamic model for aqueous electrolyte solutions at subzero temperatures, FREZCHEM2, which is a further development of the FREZCHEM model. The model uses thermodynamic data of Spencer-Møller-Weare that permit the calculation of chemical equilibria in the Na-K-Ca-Mg-Cl-SO ₄ -H ₂ O system between -60 and 25°C at atmospheric pressure. It applies the Gibbs energy minimization method for chemical equilibrium computation combined with Pitzer equations for activity coefficients and water activity calculation. The model includes both the freezing (melting) reaction pathway at fixed water amount and the evaporation (dilution) pathway at fixed temperature. The FREZCHEM2 model can be extended with respect to independent components, electrolyte species, and solids, and if corresponding thermodynamic data are available, the model may be used to compute chemical equilibria in any systems that include aqueous-solution and/or one-component solid phases.					
14. SUBJECT TERMS Aqueous solutions Computer model Freezing Chemical thermodynamics Evaporation				15. NUMBER OF PAGES 48	
				16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL		